ESAIM: PROCEEDINGS, **March 2012, Vol. 35, p. 107-113**
Fédération Denis Poisson (Orléans-Tours) et E. Trélat (UPMC), Editors

NUMERICAL VERIFICATION OF INDUSTRIAL NUMERICAL CODES

Christophe DENIS[1] and Sethy MONTAN[2]

Abstract. Several approximations occur during a numerical simulation: physical effects mapy be discarded, continuous functions replaced by discretized ones and real numbers replaced by finite-precision representations. The use of the floating point arithmetic generates round-off errors at each arithmetical expression and some mathematical properties are lost. The aim of the numerical verification activity at EDF R&D is to study the effect of the round-off error propagation on the results of a numerical simulation. It is indeed crucial to perform a numerical verification of industrial codes such as developped at EDF R&D even more for code running in HPC environments. This paper presents some recent studies around the numerical verification at EDF R&D.

Résumé. Le résultat d'un code de simulation numérique subit plusieurs approximations effectuées lors de la modélisation mathématique du problème physique, de la discrétisation du modèle mathématique et de la résolution numérique en arithmétique flottante. L'utilisation de l'arithmétique flottante génère en effet des erreurs d'arrondi lors de chaque opération flottante et des propriétés mathématiques sont perdues. Il existe à EDF R&D une activité transverse de vérification numérique consistant à étudier l'effet de la propagation des erreurs d'arrondi sur les résultats des simulations. Il est en effet important de vérifier numériquement des codes industriels et ce d'autant plus s'ils sont éxécutés dans environnements de calcul haute performance. Ce papier présente des études récentes autour de la vérification numérique à EDF R&D.

Introduction : the numerical verification at EDF R&D

Several approximations occur during a numerical simulation. For example physical effects may be discarded, continuous functions replaced by discretized ones and real numbers replaced by finite-precision representations. The use of finite-precision arithmetic generates round-off errors at each arithmetical expression and some mathematical properties are lost. For example, floating point summation is no longer associative. The same numerical code using the same data could produce different results on different computers. For example, Goel & Dash in [1] present the numerical difference obtained by running the same weather prediction code, with the same input data, on three different computer architectures. There is obviously a need to detect the effect of round-off error propagation on the computed results. Several methods have been developed over the years to analyse round-off error propagation. These include direct analysis, inverse analysis, methods based on interval arithmetic, randomised interval arithmetic and the CESTAC method.

At EDF R&D, there is a requirement to have a less intrusive tool to avoid having to rewrite the original code to study its numerical accuracy. The CADNA (Control of Accuracy and Debugging for Numerical Applications) library, developed by the Laboratoire d'Informatique de Paris 6 (http://www.lip6.fr/) appears the most promising approach for industrial applications. This paper is divided into two parts. In the first part, the

[1] EDF R&D, SINETICS Department, 1, avenue du Général de Gaulle, Clamart, France.
[2] EDF R&D, SINETICS Department, 1, avenue du Général de Gaulle, Clamart, France.

EDF R&D numerical activity is illustrated in this paper on the parallel numerical code TELEMAC-2D. The last part of this paper deals with the efficient implementation of CADNA on MPI, BLACS and BLAS librairies. This work is currently done in the context of a Phd-Thesis co-supervised by EDF and LIP6.

1. Overview of the numerical verification activity at EDF R&D

The parallel numerical code TELEMAC-2D is used in this paper to illustrate our numerical verification activity. This part is organised as follows. Firstly, the CADNA library and the parallel numerical code TELEMAC-2D are briefly presented. Then, an illustration of the Xd+P approach developped to measure the numerical quality of the computed results is shown. Last but not least, the work done around the study of the impact of the round-off error propagation in HPC is described.

1.1. The CADNA library

The CADNA (Control of Accuracy and Debugging for Numerical Applications) library, developed by the Laboratoire d'Informatique de Paris 6 (http://www.lip6.fr/), uses an implementation of discrete stochastic arithmetic (DSA) based on the CESTAC method and appears the most promising approach for industrial applications [2]. This library can be used in sequential programs written in ADA, C, C++ and Fortran. The exact result for any non-exact floating-point arithmetic operation is bounded by two consecutive floating-point values, R- and R+. The discrete stochastic arithmetic replaces the computer's deterministic arithmetic by performing each floating point operation N times, randomly rounding each time, with a probability of 0.5 to R- or R+. A typical value for N is 3. The rounding mode switch is performed through a system call. More information about the CESTAC method and the CADNA library can be found at http://www-pequan.lip6.fr/cadna/.

A translator source code has been developped at EDF R&D to implement CADNA in Fortran source code. Indeed, the CADNA library defines new datatypes for floating numbers called stochastic numbers. A stochastic number a contains three floating numbers and an integer containg the number of significant digits of a. The computing time of a program using CADNA increases as:

- the number of floating point operations is multiplied by three in contrast of the original code;
- frequent systems calls are performed to change the rounding mode;
- the number of cache defaults increases.

It is difficult to predict the overcost of computing due to points 2 and 3. The amount of memory increases at maximum of a factor 4 as a stochastic float contains one integer and three floating point numbers.

1.2. The TELEMAC-2D parallel numerical code

The TELEMAC-2D code is contained in the TELEMAC system which is developped by the EDF National Hydraulics and Environment Laboratory. The aim of TELEMAC system is to study the numerical modelling system for free surface hydrodynamics, sedimentology, water quality waves and underground flows. TELEMAC-2D solves the shallow water equations whereas TELEMAC-3D solves the full 3D free surface Navier-Stokes equations. It is mostly based on the finite element method and the basic principles of the solution procedure are detailed in [3]. The parallelism in the TELEMAC system is based on the message passing paradigm. The finite element mesh is divided into N_s subdomains $SD^{(j)}$ without finite element overlapping by using the graph partitioning tool METIS. Due to the domain decomposition, there exist nodes - called interface nodes - shared by several subdomains. Algorithm 1 summarises the two main steps of the TELEMAC-2D or TELEMAC-3D parallel version: the computation and the communication steps.

1.3. The Xd+P approach

The fields of a numerical simulation using a finite difference, finite volume or finite element method are most often represented simply by their values in xD (1-D, 2-D, or 3-D). A new approach called xD+P is introduced here in order to measure the numerical quality of the computed values, which might be seen as increasing the

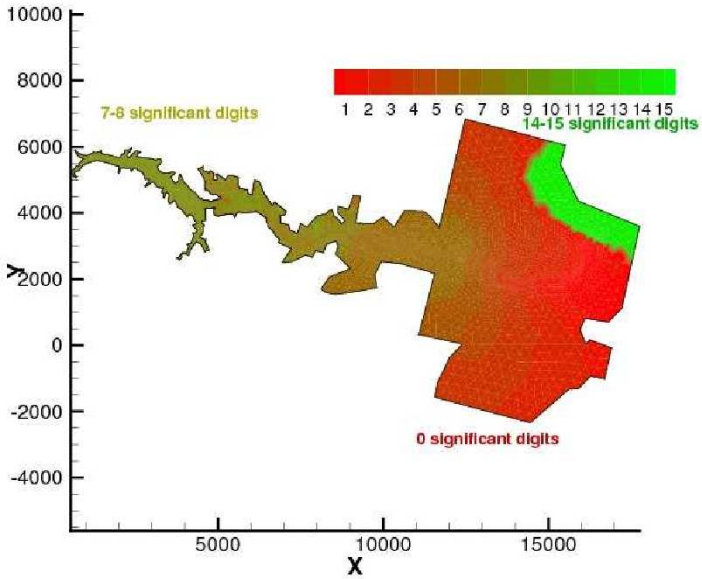
Algorithm 1 Main steps of the TELEMAC-2D parallel version

for all timestep t **do**
{**Computation step**}
New values of the nodes - including interface nodes - are locally computed in parallel on each subdomain $SD^{(j)}$. Depending of the type of computation, one to three contributions are computed and stored into one to three arrays V_1, V_2, V_3.
{**Communication step**}
The values of interfaces nodes scattered on several subdomains need to be gathered. The gather operation could be the sum, the maximum value, the minimum value or the maximum absolute value.
end for

dimension of the simulation by P. More precisely, P represents the number of decimal significant digits which are not affected by round-off errors. For instance, the approach allows the identification of potential numerical problems in portions of the mesh having P values close to zero significant digits. In addition, the xD+P approach could also be used to give confidence about the outcome of the simulation.

The xD+P approach is illustrated for a sequential numerical simulation performed in double precision with TELEMAC-2D. The case study simulates the Malpasset dam-break which occurred in the South of France in 1959. Figure 1 represents the P dimension of the water level field 25 minutes after the failure of the dam. The mesh can be divided into three main zones, depending on the numerical quality:

FIGURE 1. The Xd+p approach



- Zone 1 (in brown): the water level field is computed correctly up to 7 to 8 significant digits;
- Zone 2 (in green): the water level field is computed correctly with the maximum number of decimal digits. (The maximum number of significant decimal digits is 15 when using double precision arithmetic);
- Zone 3 (in red): the water level field is with computed with a precision of 0 significant digits.

Zone 3 highlights a potential numerical problem in the simulation. However, the water level in this region is very small, typically close to the precision of the machine. To understand why the xD+P approach detects this problem, it should be noted that the CADNA library relies on a stochastic approach based on a sample of

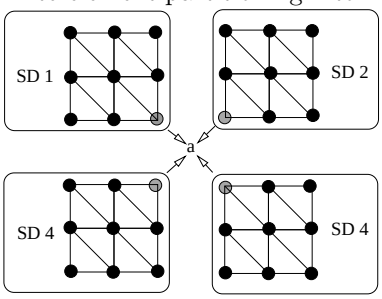
three floating point numbers rounded differently for each floating point operation in order to detect numerical instabilities. The numerical instability in Zone 3 is due to the water level value being computed as zero in the code without CADNA (with a single rounding mode). The corresponding sample computed by CADNA is (0.0, 0.0, e) where e is a small value close to the precision of the machine. Consequently, CADNA indicates there are no common digits in the sample and hence predicts 0 decimal significant digits.

1.4. **Impact of the round-off error propagation in High Performance Computing**

The round-off error propagation is exacerbated in HPC since trillions of floating arithmetic operations can be performed each second. Moreover, due to domain decomposition, the floating point arithmetic operations are not performed in the same order.

The following example illustrates this fact. Consider without loss of generality, a rectangular finite element mesh split into four subdomains as shown in Figure 2. This domain decomposition involves interface nodes shared by several subdomains.

FIGURE 2. Finite element partitioning into 4 subdomains.



Consider the gray node a shared by four subdomains. During the communication step, each subdomain receive the value of this node coming from the three other subdomains. These values will be successively added with the local value. Let a^j be the local value of the node a on $SD^{(j)}$, each subdomains compute in parallel these sums:

$$\begin{aligned} a^1 &\leftarrow a^1 + a^2 + a^3 + a^4, \quad a^2 \leftarrow a^2 + a^1 + a^3 + a^4, \\ a^3 &\leftarrow a^3 + a^1 + a^2 + a^4, \quad a^4 \leftarrow a^4 + a^1 + a^2 + a^3 \end{aligned}$$

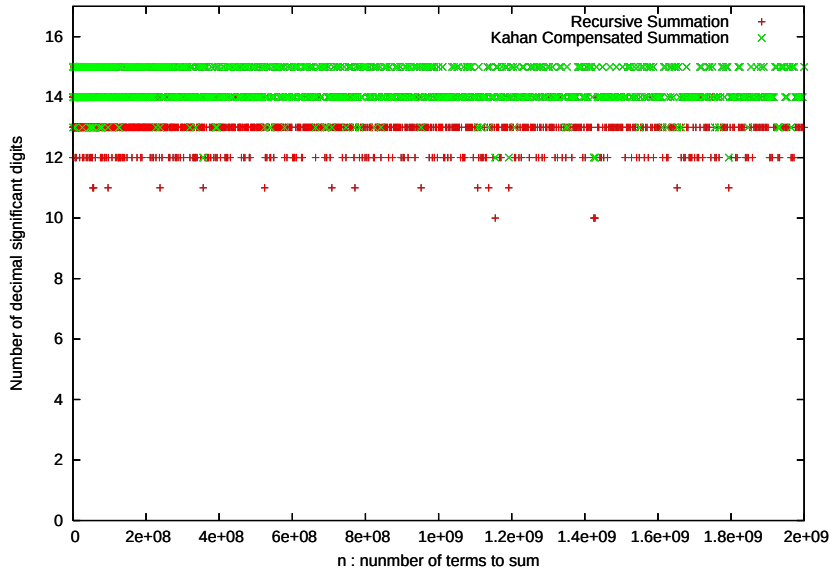
Unfortunately, as floating point computation is concerned, the result of the sums could not be the same on each subdomain due to round-off errors as the floating point sum is not associative. These differences sometimes generate deadlocks between processors. This phenomena is obviously exacerbated for simulation requiring a large amount of subdomains. The first solution given was to assign on each interfaces nodes the maximum value of the sums among subdomains. The drawbacks of this solution are the increase of the communication volume and the problem of round-off errors is hidden but still exists. Indeed, the maximum value of the sum is not necessarily the solution with few round-off errors. The communication phase has been modified in order to compute the sum in the same order among subdomains (the ascending order of MPI process). This modification allows to avoid computing the maximum value. This first solution helps to remove deadlocks between processors but unfortunately hides the numerical problem. We are now investigating compensated summation methods to increase the accuracy of the communication scheme. The Kahan summation algorithm (cf. Algorithm 2) significantly reduces the numerical error in the total obtained by adding a sequence of finite precision floating point numbers, compared to the recursive approach.

The Kahan summation algorithm has been tested with data coming from the TELEMAC-2D communication scheme. Figure 3 shows that the Kahan summation algorithm increases the number of exact significant digits.

Algorithm 2 Kahan's compensated summation method

$s \leftarrow 0.D0$
$e \leftarrow 0.D0$
for i=1,n **do**
  $tmp \leftarrow s$
  $y \leftarrow x(i) + e$
  $s \leftarrow tmp + y$
  {**Compensated term** e **computed at iteration** i **is added to** s **at iteration** $i+1$}
  $e \leftarrow (tmp - s) + y$
end for

FIGURE 3. Number of exact significant digits given by the recursive or compensated floating point summation

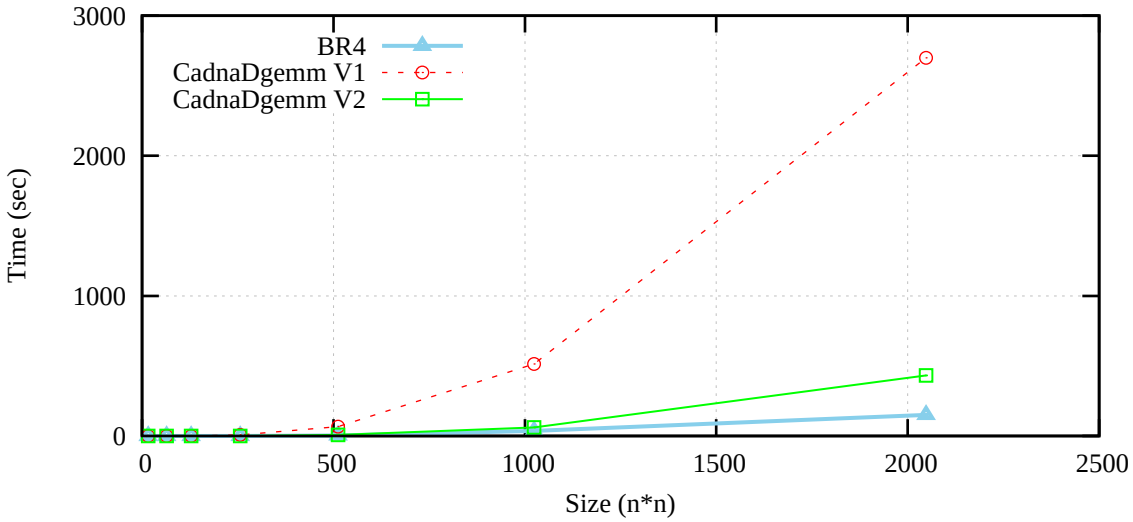


The next step is to implement a library dealing with compensated summation methods and accurate dot products to be used by EDF R&D industrial codes.

2. Implementation of CADNA in communication and scientific librairies

The CADNA library could be only used on sequential programs. Unfortunately, as previsously written in this paper, the number of floating point operations could be huge in HPC and the operations order differs from the sequential version. It is therefore important to perform numerical verification on parallel codes using libraries such as MPI, BLACS, BLAS, LAPACK. Consequently, S. Montan has developped during a trainee course at EDF R&D new libraries called CADNA_MPI and CADNA_BLACS in order to use CADNA in programs using CADNA and BLACS. CADNA_MPI has been developed for programs written in C/C++, and Fortran 90(77). It allows to exchange stochastic datas whith the MPI communication routines (point-to-point or collective) thanks to new MPI derived datatypes : MPI_SINGLE_ST for the simple precision and MPI_DOUBLE_ST for the double precision. New MPI operators have been designed to make new MPI derived datatypes compatible with the reductions routines. Theses new operators are MPI_CADNA_PROD, MPI_CADNA_SUM, MPI_CADNA_MAX,

FIGURE 4. Various DGEMM computing times.



MPI_CADNA_MIN, MPI_CADNA_MAXLOC, MPI_CADNA_MINLOC. A ping-pong communication test has been carried out in order to measure the CADNA_MPI overcost. The time to send and receive stochastic data between two processes have been measured. It is then representative of real communication time in applications. The use of new datatypes introduces a overcost of 4 compared to a floating point number. That is quite normal because a stochastic float consists of three floating point and integer. This ratio was not even originally for double precision stochastic data but by adding one integer as padding we have obtain a normal overcost. The same work on BLACS (Basic Linear Algebra Communication Subprograms) has been performed [5].

The CADNA efficient implementation in BLAS is more difficult. Consider for example the xGEMM subroutine which performs matrix multiplication (SGEMM for single precision, DGEMM for double-precision, CGEMM for complex single precision, and ZGEMM for complex double precision). xGEMM is tuned by High Performance Computing vendors to run as fast as possible, because it is the building block for so many other routines and linear solvers.

The naive way is to translate DGEMM subroutine to use CADNA double precision datatypes. This work has been done on the Netlib and the Linalg source codes [8]. These two first easier CADNA DGEMM versions were compared in terms of computing time with Atlas, Netlib and Goto2 BLAS versions. The overcost of the CADNA_DGEMM is about 1,000 which is to high to be able to simulate industrial cases. The following experiment has been carried out in order to explain this overhead. The CADNA operator overloading and the random rounding mode switch has been removed from "CadnaDgemm V1" version to obtain the so-called "CadnaDgemm V2" version. Note this latter version could be only use to explain the computing time as it is not conform to the CESTAC method. A matrix multiplication recursive algorithm based on [9] has been implemented in "CadnaDgemm V2" to obtain the "BR4" version. We have compared in terms of computing time this three versions. Indeed, Figure 4 presents the DGEMM computing time with

- the naive CADNA implementation called "CadnaDgemm V1" ;
- the "CadnaDgemm V2" implementation in which the CADNA operator overloading and call systems are removed;
- "BR4" is based on a matrix multiplication recursive algorithm in which the finest block's is a 4*4 matrix.

These results show that this significant overhead can be explained mainly by the frequent systems calls to change the rounding mode. The DGEMM algorithm needs to be redefined to reduce the overead of the impact of

changing rounding mode. This algorithm has to maximize the instruction pipeline to use efficiently streaming SIMD extensions such as SSE or AVX. This work is currently investigated in the context of a Phd-thesis co-supervised by EDF R&D and LIP6.

Conclusion and future works

This paper has presented some recent works performed at EDF R&D in the context of the numerical verification activity by using the CADNA library. A new approach called xD+P is introduced here in order to measure the numerical quality of the computed values. This approach is used to give confidence about the outcome of the simulation. The impact of the round-off error propagation in High Performance Computing has been illustrated by considering the parallel communication scheme of the industrial code TELEMAC-2D. A library dealing with accurate floating point summation and dot product to be used by EDF R&D codes is currently developped. Last but not least, the efficient implementation of CADNA in communication and scientific librairies will be achieved in the context of a Phd-thesis co-supervised by EDF R&D and LIP6.

Acknowledgments

The authors would like to thank Pr. Jean-Marie Chesneaux (University of Paris 6, LIP6) and Pr. Jean-Luc Lamotte (University of Paris 6, LIP6) for their constructive remarks on the work of Sethy Montan.

References

[1] S. Goel, and S.K. Dash. Response of model simulated weather parameters to round-off-errors on different systems. In *Environmental Modelling and Software*, pages 1164–1174, 2007

[2] Fabienne Jézéquel, , Jean-Marie Chesneaux, and Jean-Luc Lamotte. A new version of the CADNA library for estimating round-off error propagation in Fortran programs. In *Computer Physics Communications*, volume 181(11), pages 1927–1928, 2010.

[3] Jean-Michel Hervouet. Hydrodynamics of Free Surface Flows: Modelling with the Finite Element Method. In *John Wiley & Sons*, Chichester, 2007.

[4] The TELEMAC system, http://www.telemacsystem.com.

[5] Jack J. Dongarra, R.Clint Whaley. LAPACK Working Note 94: A User"s Guide to the BLACS v1.0. University of Tennessee, 1995.

[6] Basic Linear Algebra Technical Forum. Basic Linear Algebra Technical Forum Standard. University of Tennessee, 2001.

[7] Susan Blackford, Jack J. Dongarra LAPACK Working Note 41: Installation Guide for LAPACK, Version 3.0. University of Tennessee, Tennessee,1999.

[8] Philippe Trebuchet. The linalg Library. http://www-apr.lip6.fr/ trebuche/linalg.html

[9] Peter Gottschling, David S. Wise, Adwait Joshi. Generic support of algorithmic and structural recursion for scientific computing. In *International Journal of Parallel, Emergent and Distributed Systems*, volume 24(6), pages 479–503, 209.